

IN THE SPECIFICATION

(1) In the Abstract, please replace the entire paragraph with the following rewritten paragraph:

A computer system (e.g., a personal computer (PC)) is loaded with multiple versions of the bootable program (e.g., an operating system (OS)). The boot record for each bootable program version is hashed to produce a digest and the digest is signed using the cryptographic signature engine using the program's private installation key. The resulting signature, ~~is stored~~ along with data indicating the program's name and version, is stored in fields of the non-volatile memory. When the system boots with a version of the program, the active entry is decrypted and the resulting data is compared. If a new version is being booted, then it is determined if the new version is the alternative entry. If so, the active entry is discarded and the alternative version is moved to be the active entry and the system boots with the new version of the program.

(2) In the Detailed Description, please replace the paragraph beginning at page 14, line 6 and ending at page 14, line 21 with the following rewritten paragraph:

If the result of the test in step 605 is YES, then in step 608 the signature in the primary table entry is compared to the boot record signature in the active partition. If the result of the compare is YES, then in step 609 the primary and alternate entries in the version management table of the non-volatile memory 102 are locked. A branch is then taken to step 607 where the OS in the active partition is booted. If the result of the test in step 608 is NO, then action has been taken to switch to a new version of the OS. In step 610, the signature of the alternate table entry is compared to the BR in the active partition. If they do not compare then in step 611, the boot process is halted and action must be taken to correct the system configuration before proceeding. If they do compare in step 610, then the alternate entry has been selected as the new version of the OS. In step 612, the primary entry is cleared and the contents of the alternate entry is moved to the primary entry. This effectively prevents the version defined by the primary entry from being used again. If a third entry is present, then the contents of the third entry are

moved to the alternate entry in step 613. Whether or not a third entry is valid in step 613, a branch is taken to step 609 where the primary and alternate entries in the version management table are locked. The third entry is left unlocked for use by the OS loader.

(3) In the Detailed Description, please replace the paragraph beginning at page 15, line 8 and ending at page 15, line 28 with the following rewritten paragraph:

If the result of the test in step 704 is NO, then in step 708 a test is done to determine if a version management table is present in non-volatile memory 102. If the result of the test in step 708 is NO, then in step 709 only one OS version has been loaded in the system 200 and all of the possible entries for the version management table are left unlocked for later use by the OS loader. In step 710, the OS in the active partition is booted. If the result of the test in step 708 is YES, then in step 711 the signature in the primary table entry is compared to the boot record signature in the active partition. If the result of the compare is YES, then in step 712 the primary and alternate entries in the version management table of the non volatile memory 102 are locked. A branch is then taken to step 710 where the OS in the active partition is booted. If the result of the test in step 711 is NO, then action has been done to switch to a new version of the OS. In step 713, the signature of the alternate table entry is compared to the BR in the active partition. If they do not compare then in step 714, the boot process is halted and action must be taken to correct the system configuration before proceeding. If they do compare in step 713, then the alternate entry has been selected as the new version of the OS. In step 715, the primary entry is cleared and the contents of the alternate entry is moved to the primary entry. This effectively prevents the version defined by the primary entry from being used again. If a third entry is present, then the contents of the third entry are moved to the alternate entry in step 716. Whether or not a third entry is valid in step 716, a branch is taken to step 712 where the primary and alternate entries in the version management table are locked. The third entry is left unlocked for use by the OS loader.